UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/970,312	10/02/2001	Kuldeep K. Dhar	1194.12-0002	9682

164 7590 06/14/2007

KINNEY & LANGE, P.A.
THE KINNEY & LANGE BUILDING
312 SOUTH THIRD STREET
MINNEAPOLIS, MN 55415-1002

EXAMINER	
DESHPANDE, KALYAN K	
ART UNIT	**PAPER NUMBER**
3623	

MAIL DATE	DELIVERY MODE
06/14/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/970,312	DHAR ET AL.	
	Examiner	Art Unit	
	Kalyan K. Deshpande	3623	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 March 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-14 and 21-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-14 and 21-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *10/2/2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *1/16/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Introduction

1. The following is a final office action in response to the communications received on March 19, 2007. Claims 1-14 and 21-27 are now pending in this application.

Examiner's Note

2. Examiner appreciates Applicants' noting of the typographical error and Applicants are correct in their assumes. The typographical error has been corrected in this action.

Response to Amendment

3. Applicants' amendments to claims 1, 8, 22, 25, and 27. Examiner maintains all previously asserted grounds of rejection.

Response to Arguments

4. Applicants' arguments filed on September 25, 2006 have been fully considered but are moot in view of the new ground(s) of rejection. Applicants argue Bengston, Nichols, and Bacon fail to teach tasks are defined prior to runtime setup.

In response to Applicants argument Bengston, Nichols, and Bacon fail to teach tasks are defined prior to runtime setup, Examiner respectfully disagrees. Bengston clearly teaches tasks are defined prior to runtime setup (see column 4 lines 45-65, column 6 lines 26-67, column 7 lines 30-58, column 9 lines 1-33, and figures 3, 6, and 7; where tasks are defined in the workflow definitions prior to execution of the workflow, which is the same as prior to runtime setup.). Examiner is confused as to Applicants' arguments. Applicants admit that Bengston and Nichols teach defining tasks prior to

runtime (see Remarks page 23) and Bacon teaches dynamically modifying workflows at runtime (see Remarks page 23). Applicants claim that in the present invention tasks are defined prior to workflow execution, however, argue the references fail to teach tasks are defined at runtime (see Remarks page 23). Examiner is confused as to contradiction between the claimed language and the recited argument. Furthermore, the recited claims call for the *tasks* to be defined prior to runtime, however, Applicants appear to be arguing that the *workflow* be defined at or prior to runtime. Examiner would greatly appreciate clarification as to exactly what is being claimed.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 1-14 and 21-27 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Independent claims 1, 8, 22, and 25 recite the limitation "differentiated tasks defined prior to runtime setup". First, it is unclear as to the exactly meaning of "runtime setup". For the purposes of examination, Examiner interprets "runtime setup" to mean a time during runtime, i.e. a runtime mode. Second, the recited claim calls for tasks being defined "prior to runtime setup", however, the presented arguments argue that the references fail to teach tasks defined "during setup" (see Remarks p. 23). Applicants admit that Bengston and Nichols teach defining tasks prior to runtime (see Remarks page 23) and Bacon teaches dynamically modifying workflows at runtime (see Remarks

page 23). Applicants claim that in the present invention tasks are defined prior to workflow execution, however, argue the references fail to teach tasks are defined at runtime (see Remarks page 23). Examiner is confused as to contradiction between the claimed language and the recited argument. Furthermore, the recited claims call for the *tasks* to be defined prior to runtime, however, Applicants appear to be arguing that the *workflow* be defined at or prior to runtime. For the purposes of examination, Examiner is interpreting this limitation to mean that tasks are defined prior to runtime.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-14 and 21-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bengston (U.S. Patent No. 6728947) in view of Nichols et al. (U.S. Patent No. 6018730) and in further view of Bacon et al. (U.S. Patent No. 6430538).

As per claim 1, Bengston teaches:

A workflow management system for hosting process-based tasks and decisioning, the workflow management system comprising:

a compiled program kernel containing multiple differentiated tasks defined prior to runtime setup as separate functions with the compiled program (see column 4 lines 45-65, column 6 lines 26-67, column 7 lines 30-58, column 9 lines 1-33, and figures 3, 6, and 7; where the system processes steps and functions of

a workflow. The reference invention is a software component enabling the processing of the workflow. Figure 3 displays one of the plurality of functions that are part of the reference invention. Each workflow definition is created by linking nodes, where each node represents a different task. The linking between the nodes can be simple or complex and dependant. The workflow definition can be created prior to execution, where execution is the same as runtime.);

a graphical interface having a list of geometric shapes and a workspace, each geometric shape being an abstracted object-based representation of functions within the compiled program kernel, the workspace for organizing and linking multiple geometric shapes in a sequential arrangement of objects, the sequential arrangement of objects corresponding to an order in which the multiple differentiated tasks are performed by the compiled program kernel (see column 6 lines 26-67, column 7 lines 1-29, and figure 2; where a workflow designer can create or edit a workflow. The functions are represented by icons. The designer can create a workflow by organizing and assembling the icons. The designers screen is split, where one portion is reserved for displaying the plurality of icons and the remaining portion available to be used as workspace. The designer can drag-and-drop the icons on to the workspace to create the workflow.); and

a database for storing the arrangement of objects as a checklist (see column 9 lines 55-67, column 10 lines 1-34, and figures 2 and 3; where processes and workflows can be saved to RAM or to disk in various forms.).

Bengston fails to explicitly teach "a data dictionary defining discrete data elements accessible by the compiled program kernel to process the checklist, wherein the contents of the data dictionary are specific to a selected industry", "a database for storing entry conditions that are associated with each of the multiple differentiated tasks", and "wherein the entry conditions are evaluated by the compiled program kernel with respect to each of the multiple differentiated tasks such that a particular one of the multiple differentiated tasks is performed only if all of the entry conditions associated with that particular one of the multiple differentiated tasks evaluate to true". Nichols in an analogous art teaches "a data dictionary defining discrete data elements accessible by the compiled program kernel to process the checklist, wherein the contents of the data dictionary are specific to a selected industry" (see column 21 lines 31-67, column 22 lines 1-40, and column 59 line 25 – column 64 line 45; where a data dictionary is disclosed. The data dictionary has a domain component that defines the data dictionary as related to a specific industry. The examples provided include insurance underwriting and credit lending). Bacon, in an analogous art, teach "a database for storing entry conditions that are associated with each of the multiple differentiated tasks" (see column 5 lines 23-48, column 9 lines 27-38, and column 10 lines 27-40; where entry conditions associated with activities are stored in a database.), and "wherein the entry conditions are evaluated by the compiled program kernel with respect to each of the multiple differentiated tasks such that a particular one of the multiple differentiated tasks is performed only if all of the entry conditions associated with that particular one of the multiple differentiated tasks evaluate to true" (see column 5 lines 23-48, column 9 lines

27-38, and column 10 lines 27-40; where entry conditions are used to evaluate the next activity in the workflow process. The set of conditions are evaluated as true/false conclusions and a particular activity is determined to be performed based on the conclusion of the true/false assessment.). The advantage of using these features are that they streamline the use of automated tools reducing overall costs. It would have been obvious, at the time of the invention, to one of ordinary skill in the art to combine the feature of "a data dictionary defining discrete data elements accessible by the compiled program kernel to process the checklist, wherein the contents of the data dictionary are specific to a selected industry", "a database for storing entry conditions that are associated with each of the multiple differentiated tasks", and "wherein the entry conditions are evaluated by the compiled program kernel with respect to each of the multiple differentiated tasks such that a particular one of the multiple differentiated tasks is performed only if all of the entry conditions associated with that particular one of the multiple differentiated tasks evaluate to true" taught by Nichols to the Bengston system in order to streamline the automation of processes thereby reducing overall costs, which is a goal of Bengston (see column 1 lines 55-67 and column 2 lines 1-4).

As per claim 2, Bengston teaches:

The workflow management system of claim 1, further comprising:

administrative tools for accessing a stored checklist, the administrative tools capable of altering parameters associated with each geometric shape in the stored checklist (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a

process editing tool allows a user to configure the parameters of the steps and functions of the workflow.).

As per claim 3, Bengston teaches:

The workflow management system of claim 1, wherein multiple checklists may be stored in the database (see column 9 lines 55-67, column 10 lines 1-34, and figures 2 and 3; where processes and workflows can be saved to RAM or to disk in various forms.).

As per claim 4, Bengston teaches:

The workflow management system of claim 1, wherein the graphical interface permits dynamic alteration of the ordered arrangement of objects in the stored checklist without restarting the system and without recompiling the compiled program kernel (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a the process editing tool allowing a user to configure the parameters of the steps and functions of the workflow is web-enabled and built using java. The system creates new files with the workflow definitions provided by the designer. The creation of files does not require that the system, application server, or web server be restarted.).

As per claim 5, Bengston teaches:

The workflow management system of claim 1, wherein the graphical interface is web-enabled, such that a remote user can access the graphical interface to modify the ordered arrangement of objects in the stored checklist (see column 4 lines 66-67, column 5 lines 1-45, column 6 lines 4-25, and figure 1 and 2; where the system uses a communications channel that is connected to the Internet. The editing device

can be enabled in any web browser such as Netscape Navigator or Internet Explorer.).

As per claim 6, Bengston teaches:

The workflow management system of claim 1, further comprising:

an automated messaging system for communicating action items with registered users in the system, the messaging system being Internet-based (column 14 lines 16-40; where parameters can be adjusted such that email message, a phone call, or a paging message can be sent to defined users regarding the status of action items of the workflow.).

As per claim 7, Bengston teaches:

The workflow engine of claim 6, wherein the automated messaging system includes electronic mail (column 14 lines 16-40; where parameters can be adjusted such that email message, a phone call, or a paging message can be sent to defined users regarding the status of action items of the workflow.).

As per claim 8, Bengston teaches:

A workflow system for programmatically managing dynamic workflow processes, the workflow system comprising:

a workflow engine for performing task list processing, the workflow engine being a software component containing a plurality of discrete functions defined prior to runtime setup (see column 4 lines 45-65, column 6 lines 26-67, column 7 lines 30-58, column 9 lines 1-33, and figures 3, 6, and 7; where the system processes steps and functions of a workflow. The reference invention is a

software component enabling the processing of the workflow. Figure 3 displays one of the plurality of functions that are part of the reference invention. Each workflow definition is created by linking nodes, where each node represents a different task. The linking between the nodes can be simple or complex and dependant. The workflow definition can be created prior to execution, where execution is the same as runtime.); and

a workflow designer for configuring task lists, the workflow designer having an object-based interface for drag-and-drop creation of task lists, the workflow designer having a display window divided into a function list and a workspace, the function list containing multiple symbols, each symbol corresponding to at least one of the plurality of discrete functions within the workflow engine, the workspace providing a graphical area for assembly of ordered task lists, the workflow designer allowing for assembly of ordered tasks by dragging and dropping one of the multiple symbols into the workspace, the workflow designer provides graphical links for assembling an ordered task list from multiple discrete symbols (see column 6 lines 26-67, column 7 lines 1-29, and figure 2; where a workflow designer can create or edit a workflow. The functions are represented by icons. The designer can create a workflow by organizing and assembling the icons. The designers screen is split, where one portion is reserved for displaying the plurality of icons and the remaining portion available to be used as workspace. The designer can drag-and-drop the icons on to the workspace to create the workflow.);

wherein the workflow engine performs discrete functions in an order determined by the ordered task list (see column 4 lines 45-65, column 6 lines 26-67, column 7 lines 1-29, and figures 2, 3, 6, and 7; where the system performs the discrete functions ordered and listed in the workflow by the workflow designer.).

Bengston does not explicitly teach a method to "render a financial offer decision" in the financial services industry, "entry conditions to be defined and associated with any of the plurality of discrete functions, wherein each entry condition is evaluated by the workflow engine with respect to each of the plurality of discrete functions such that a particular one of the plurality of discrete functions is executed by the workflow engine only if all of the entry conditions are associated with that particular one of the plurality of discrete functions evaluate to true", "a data dictionary defining discrete data elements and data relationships that are associated with each of the plurality of discrete functions of the workflow engine, wherein the contents of the data dictionary are specific to the selected industry", and "wherein the workflow engine performs discrete functions for which all associated entry conditions evaluate to true". The limitations "entry conditions to be defined ans associated with any of the plurality of discrete functions, wherein each entry condition is evaluated by the workflow engine with respect to each of the plurality of discrete functions such that a particular one of the plurality of discrete functions is executed by the workflow engine only if all of the entry conditions are associated with that particular one of the plurality of discrete functions evaluate to true", "a data dictionary defining discrete data elements and data relationships that are associated with each of the plurality of discrete functions of the workflow engine, wherein the

contents of the data dictionary are specific to the selected industry", and "wherein the workflow engine performs discrete functions for which all associated entry conditions evaluate to true" are already addressed by the rejection of claim 1; therefore the same rejection applies to this claim. Although Bengston fails to disclose a method to "render a financial offer decision" in the financial services industry, Bengston discloses a workflow system that renders a solution by solving a defined set of ordered tasks that can be applied to a variety of industries, regardless of the intended field of use of the method. Bengston teaches a workflow system for the printing industry, though the system has utility in other applications (see column 16 lines 28-33). The system being adapted to a financial services industry, specifically to render a financial offer, is irrelevant since the intended use does not change the overall functionality of the system. The intended use must result in a manipulative difference as compared to the prior art. The intended use must result in a manipulative difference as compared to the prior art. See *In re Casey*, 152 USPQ 235 (CCPA 1967) and *In re Otto*, 136 USPQ 458, 459 (CCPA 1963). Therefore, it would have been obvious, at the time of the invention, to one of ordinary skill in the art to use the Bengston system at a financial services industry, specifically to render a financial offer, because Bengston system is designed to be used in a workflow system to render a solution by solving a defined set of ordered tasks regardless of the intended use.

As per claim 9, Bengston teaches:

The workflow system of claim 8, wherein the workflow designer is Internet-based and wherein the function list and the workspace are accessible using an Internet

browser (see column 4 lines 66-67, column 5 lines 1-45, column 6 lines 4-25, and figure 1 and 2; where the system uses a communications channel that is connected to the Internet. The editing device can be enabled in any web browser such as Netscape Navigator or Internet Explorer.).

As per claim 10, Bengston teaches:

The workflow system of claim 8, further comprising:

a workflow setup utility for configuring parameters within the checklist (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a process editing tool allows a user to configure the parameters of the steps and functions of the workflow.).

As per claim 11, Bengston teaches:

The workflow system of claim 10, wherein the workflow setup utility is web-enabled (see column 7 lines 30-58; where the process editing tool is a java applet that can be embedded in a web browser.).

As per claim 12, Bengston teaches:

The workflow system of claim 8, further comprising:

a messaging system for programmatically prompting a user to take action (see column 13 lines 45-67; where the system prompts the workflow designer to input the observing addresses).

As per claim 13, Bengston teaches:

The workflow system of claim 12, wherein the messaging system generates a digital message (see column 13 lines 45-67; where the system prompts the workflow

designer to input the observing addresses. The prompt to the user is a digital message on the screen.).

As per claim 14, Bengston teaches "the workflow system can be configured using the process editing tool to execute processes determined by the workflow designer" (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a process editing tool allows a user to configure the parameters of the steps and functions of the workflow.). Bengston fails to teach the messaging system forwards a document to the user for review and action. Though Bengston does not explicitly teach the messaging system forwarding a document to the user for review and action, the Bengston system can be configured to produce such an action. The Bengston system provides a process editing tool which allows a user to specify the parameters of the process steps defined in the workflow. Bengston provides the example of sending a paging message to a specific user by appropriately defining the process steps (see column 8 lines 3-22). Furthermore, Bengston teaches intervening manual steps in a workflow process (see column 1 lines 33-46). The advantage of forwarding a document to a user for review and action is to facilitate human intervening steps in the workflow. Thus, it would have been obvious, at the time of the invention, to one of ordinary skill in the art to appropriately configure the Bengston system to forward a document to a user for review and action in order to facilitate the intervening human step in the workflow, which is a goal of Bengston (see column 1 lines 33-46).

As per claim 21, Bengston fails to explicitly teach a data dictionary specific to the lending industry. This limitation is addressed by the rejection of claim 1; therefore the same rejection applies to this claim.

As per claim 22, Bengston teaches:

A system for programmatically rendering a process-based decision, the system comprising:

administrative tools for creating process categories and checklists associated with each process and discrete tasks defined prior to runtime setup and for modifying decision parameters in each checklist (see column 6 lines 26-67, column 7 lines 5-67, column 8 lines 1-37, column 9 lines 1-33, and figure 2; where a process editing tool allows a user to configure the parameters of the steps and functions of the workflow. This can be done prior to execution of the workflow, which is the same as prior to runtime.);

a decision database for storing the process categories, the checklists and the selection criteria (see column 9 lines 55-67, column 10 lines 1-34, and figures 2 and 3; where processes and workflows can be saved to RAM or to disk in various forms.);

a workflow engine for automatically processing input from a remote user and generating an instant decision based on the checklist, wherein the workflow engine capable of securely transmitting the instant decision to the remote user, and wherein the workflow engine capable of brokering communications between the remote user and a process administrator associated with the instant decision (see column 11

lines 56-67, column 12 lines 1-24, and column 14 lines 16-40; where processing devices receive an input from another processing device or from the initiating device. The processing device evaluates the information in the workflow file and executes any necessary steps. The processing device then transmits is evaluation to the necessary resource (e.g. if the workflow input dictates that the processing device print information, this decision and the information to print is sent to the printer, if the workflow input dictates that the next process step needs to be executed by another device, the processing device transmits all necessary information to the following device for further processing). Parameters can be adjusted such that email message, a phone call, or a paging message can be sent to defined users regarding the status of action items of the workflow.); and

a messaging system for routing two-way communications between the remote user and the process administrator, the messaging system providing a digital record of programmatic transactions (column 14 lines 16-40; where parameters can be adjusted such that email message, a phone call, or a paging message can be sent to defined users regarding the status of action items of the workflow.).

Bengston fails to explicitly teach the modifying and processing of entry conditions associated with teach task, where the entry conditions are defined on a checklist and the evaluation of the entry conditions determine whether a task will be executed as recited in claim 22, however, these limitations are already addressed by the rejection of claim 1; therefore the same rejection applies to this claim. Bengston also fails to explicitly teach a data dictionary for defining data elements and data relationships

specific to a selected industry. These limitations are also already addressed by the rejection of claim 1; therefore the same rejection applies to this claim. Claim 22 further recites the limitation that the data dictionary "provides a dynamic fetch and store interface with the decision database, and wherein the dynamic data dictionary is configured to provide, translate, and modify data presentation with respect to both the remote user and the workflow engine" that Bengston fails to explicitly teach. Nichols, in an analogous art, teaches a data dictionary that "provides a dynamic fetch and store interface with the decision database, and wherein the dynamic data dictionary is configured to provide, translate, and modify data presentation with respect to both the remote user and the workflow engine" (see column 59 line 25 – column 64 line 45; where a data dictionary is disclosed. The data dictionary has a domain component that defines the data dictionary as related to a specific industry. The examples provided include insurance underwriting and credit lending. Furthermore, the data is organized in conjunction with a database that is operable to fetch and store data objects. Data can also be retrieved in order to be modified or translated.). The advantage of using these features are that they streamline the use of automated tools reducing overall costs. It would have been obvious, at the time of the invention, to one of ordinary skill in the art to combine the feature of "provides a dynamic fetch and store interface with the decision database, and wherein the dynamic data dictionary is configured to provide, translate, and modify data presentation with respect to both the remote user and the workflow engine" taught by Nichols to the Bengston system in order to streamline the automation of processes thereby reducing overall costs, which is a goal of Bengston (see column 1

lines 55-67 and column 2 lines 1-4). Claim 22 further recites that the data dictionary be formatted in XML. Examiner takes Official Notice that is old and well-known in the art to use XML for dynamic data transactions. The advantage of using XML for the data dictionary is that it facilitates dynamic data transactions thereby reducing automation time, resources, and expenses. It would have been obvious, at the time of the invention, to one of ordinary skill in the art to format the data dictionary in XML in order to streamline automation of processes and transactions in order to reduce overall costs, which is a goal of Bengston (see column 1 lines 55-67 and column 2 lines 1-4).

As per claim 23, Bengston teaches:

The system of claim 22, further comprising:

a user interface for entering user information (see column 8 lines 3-22; where a user can set the parameters for a process definition where the parameters include user information such as personal identification number, pager number, phone number, and email address.).

As per claim 24, Bengston teaches:

The system of claim 22, wherein entry conditions and the selection criteria associated with the checklist are modified and new checklists are created dynamically without restarting the system (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a the process editing tool allowing a user to configure the parameters of the steps and functions of the workflow is web-enabled and built using java. The system creates new files with the workflow definitions provided by the

designer. The creation of files does not require that the system, application server, or web server be restarted.).

As per claim 25, Bengston teaches:

A method for workflow processing and programmatic decision-making based on object-based processes stored in memory, the method comprising:

Defining a plurality of differentiated tasks at a pre-runtime software design stage (see column 6 lines 26-67, column 7 lines 5-67, column 8 lines 1-37, column 9 lines 1-33, and figures 1-3; where tasks are associated with nodes Each task is defined prior to execution, which is the same as a pre-runtime software design stage.);

receiving input from a remote source (see column 10 lines 62-67, column 11 lines 1-67, column 12 lines 1-67, column 13 lines 1-9, and figures 1, 4, 6, and 7; where the initiating device transmits a request and the workflow file to the first processing device. The first processing device receives the input from the initiating device.);

determining programmatically an input type according to the received input (see column 10 lines 62-67, column 11 lines 1-67, column 12 lines 1-67, column 13 lines 1-9, and figures 1, 4, 6, and 7; where processing devices scan the workflow file to determine which type of procedure steps need to be executed.);

retrieving automatically a stored process checklist from a decision database according to the input type (see column 10 lines 62-67, column 11 lines 1-67, column 12 lines 1-67, column 13 lines 1-9, and figures 1, 4, 6, and 7; where the

workflow file contains specific process steps and these steps (input types) dictate to the system which set of instructions (checklist) to execute.);

processing programmatically the received information based on parameters associated with the stored process checklist (see column 10 lines 62-67, column 11 lines 1-67, column 12 lines 1-67, column 13 lines 1-9, and figures 1, 4, 6, and 7; where the workflow file contains specific process steps and these steps (input types) dictate to the system which set of instructions (checklist) to execute.);

rendering an automatic decision based on the processed received information (see column 10 lines 62-67, column 11 lines 1-67, column 12 lines 1-67, column 13 lines 1-9, and figures 1, 4, 6, and 7; where the processing device renders a decision as to the next step of the workflow. For example, the processing device can execute the subsequent process step, can automatically write information to the workflow file, or pass the workflow file on to the next processing device.); and

communicating programmatically the automatic decision to the remote source (column 14 lines 16-40; where parameters can be adjusted such that email message, a phone call, or a paging message can be sent to defined users regarding the status of action items of the workflow.).

Bengston fails to explicitly teach the use of entry conditions associated with tasks that are evaluated in order to determine if the task should be executed and the use of a data dictionary to store a process checklists and the entry conditions for a specific industry. These limitations are already addressed by the rejection of claim 1; therefore the same rejection applies to this claim..

As per claim 26, Bengston teaches:

The method of claim 25, wherein the step of processing comprises:

querying a remote server for information related to the received input (see column 11 lines 56-67 and column 12 lines 1-24; where the processing device retrieves a file over the communication channel for information dictated by the input).;

receiving a response containing the set of data elements and data relationships related to received input (see column 11 lines 56-67 and column 12 lines 1-24; where the processing device receives the file requested. The information file can also be automatically sent to the requesting device without the requesting device making a request.); and

evaluating the received response and the received input according to the entry conditions associated with the stored process checklist (see column 11 lines 56-67 and column 12 lines 1-24; where the processing device uses the requested information to process the workflow according to the parameters defined by the work flow designer.).

Bengston fails to explicity teach the features of a "data dictionary" and the evaluation of entry conditions. These limitations are already addressed in the rejection of claim 1; therefore the same rejection applies to this claim.

As per claim 27, Bengston teaches:

The method of claim 25, wherein before receiving, the method comprises:

creating a process checklist using an administrative utility, the administrative utility having an object-based, graphical interface wherein an authorized user creates a workflow process by dragging and dropping the defined tasks into a workspace and linking tasks into order (see column 6 lines 26-67, column 7 lines 1-29, and figure 2; where a workflow designer can create or edit a workflow. The functions are represented by icons. The designer can create a workflow by organizing and assembling the icons. The designers screen is split, where one portion is reserved for displaying the plurality of icons and the remaining portion available to be used as workspace. The designer can drag-and-drop the icons on to the workspace to create the workflow.);

configuring parameters associated with each task in the process checklist using the administrative utility (see column 7 lines 5-67, column 8 lines 1-37, and figure 2; where a process editing tool allows a user to configure the parameters of the steps and functions of the workflow.); and

storing the process checklist in a decision database (see column 9 lines 55-67, column 10 lines 1-34, and figures 2 and 3; where processes and workflows can be saved to RAM or to disk in various forms.).

Bengston fails to explicity teach the features of a "data dictionary" and the evaluation of entry conditions. These limitations are already addressed in the rejection of claim 1; therefore the same rejection applies to this claim.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kalyan K. Deshpande whose telephone number is (571)272-5880. The examiner can normally be reached on M-F 8am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tariq Hafiz can be reached on (571) 272-6729. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

kkd

Beth Van Doren
Beth Van Doren
AU 3623
Primary Examiner